ARMY RESEARCH LABORATORY

Design and Testing of an Interference Modulator for DEW Applications

by David K. Abe, Andrea K. Mark, Alan Bromborsky,
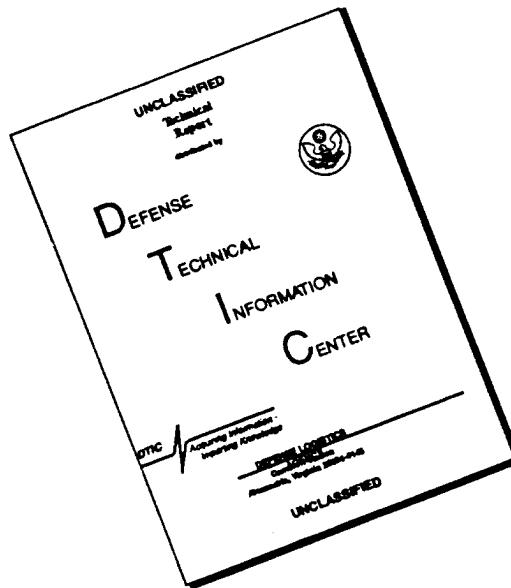Carl J. Lazard, and Romeo del Rosario

DTIC QUALITY INSPECTED 4

19960930 107

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Army Research Laboratory

Adelphi, MD 20783-1197

ARL-TR-1049   **September 1996**

Design and Testing of an Interference Modulator for DEW Applications

David K. Abe
Weapons Technology Directorate

Andrea K. Mark
Information Science and Technology Directorate

Alan Bromborsky, Carl J. Lazard, Romeo del Rosario
Weapons Technology Directorate

Abstract

The interference modulator (IM) is a microwave source concept that can produce a broadband, amplitude modulated radio frequency (rf) waveform with independently tunable carrier and modulation frequencies. In its simplest form, the IM is a two-channel device that can produce an amplitude modulated signal at 50-percent duty by either generating and beating two electromagnetic waves together in the far field or by combining the two signals with a four-port hybrid network. With additional channels, an IM can be made to synthesize more complex waveforms and duty factors. A general methodology for the design of an n-channel ($n \geq 2$) IM is developed. As a validation of the design methodology, a two-channel prototype IM was constructed. The design, fabrication, and testing of the two-channel prototype are described in detail. Experiments carried out with the prototype IM have successfully demonstrated the IM concept, producing a source with a modulation envelope that is continuously tunable over four decades (from 100 Hz to 1 MHz), with a carrier frequency that is continuously tunable from 800 MHz to 2 GHz.

Contents

List of Figures

List of Tables

1 Introduction

The interference modulator (IM) is a microwave source that can generate a broadband, amplitude-modulated radio frequency (rf) waveform with independently tunable carrier and modulation frequencies. Two key features of the IM are (1) cw operation of the rf source components, which minimizes the power conditioning requirements, and (2) operation over a very broad bandwidth. These features make the IM a highly flexible microwave source with applications in studies of electromagnetic (EM) coupling and the susceptibility of electronic components and systems. This report presents the methodology for the design of an IM and describes experiments carried out with a prototype.

1.1 Motivation for Development of IM

EM radiation may couple into a complex object through a variety of paths. Front-door coupling takes place when an externally applied signal enters the system along an intentional path for EM radiation, such as an antenna or sensor aperture. Back-door coupling takes place when the radiation enters the system along an unintentional path, such as cables or cracks in the system housing.

Amplitude-modulated rf signals can be more effective at jamming or upsetting electronic systems than non-amplitude-modulated signals. Amplitude modulation could contribute to effective electronic upset in the following manner: the high-frequency (short wavelength) carrier could allow the rf signal to enter the system via a front- or back-door path, such as a sensor port or a seam in the hardware enclosure; once inside the target system, the lower frequency modulation (if properly selected) could be rectified by nonlinear electronic circuit elements, such as transistor junctions, and be erroneously interpreted as valid data by the system logic. If this occurs, a failure or upset of the target system may result.

Significant changes in the efficiency of rf coupling into enclosures can be generated by small changes in rf frequency or modulation. Coupling effectiveness may change by as much as 20 to 30 dB for changes in frequency as small as a few megahertz. Because of this sensitivity, the ideal source for evaluating the rf susceptibility of a given system and/or potentially upsetting it should have a carrier frequency and modulation envelope with independent and near-infinite tunabilities.

Commercially available components do not have the desired bandwidth and/or tunability for these applications. The IM is an attempt at a practical design of a device with a widely selectable carrier frequency and independently tunable modulation envelope.

1.2 IM Basics

Two examples of IM configurations are shown conceptually in figure 1. A high-frequency carrier signal, f_c, and a lower frequency modulation signal, f_m, are the inputs to the IM. For outputs, the IM will pass the carrier frequency, f_c, and generate a processed signal consisting of the carrier frequency plus (or minus) the modulation frequency, $f_c + f_m$ ($f_c - f_m$). These two output signals can be post-amplified and radiated into free space through a pair of matched antennas. In the far field, f_c and $f_c + f_m$ ($f_c - f_m$) will linearly add to produce an amplitude-modulated EM wave with a 50-percent duty factor and a peak amplitude that is twice that of the individual waves.

To illustrate the modulation effect of linearly combining two waves closely spaced in frequency, figure 2 presents a plot of two sinusoidal waves with equal amplitudes, one with a frequency of 1.0 GHz, and the other with a frequency of 1.1 GHz. The sum of the two waveforms is plotted in figure 3, demonstrating the envelope modulation of the carrier frequency. Note that the peak amplitude of the sum is twice that of the constituent waveforms. The spectral content of this "beat wave" is a superposition of the frequency spectra of each of the constituent waves, as shown in figure 4, the Fourier transform of figure 3. The broadening of the spectrum in figure 4 is a consequence of the finite time duration of the wave used in the numerical analysis.

There are several advantages to producing an amplitude-modulated signal in this fashion, including minimization of the system volume and weight, reduced stress on individual system components, and modularity. By generating the modulation envelope through the interference of two waves in the far field (see fig. 1(b) and (c)), we can operate the two rf sources in a cw mode where the individual sources can operate at one-half the average power (one-fourth the peak power) of the equivalent single-antenna, amplitude-modulated source. The lower power requirements of the IM concept result in reduced voltage stress on the individual components (amplifiers, antenna interfaces, transmis-

4
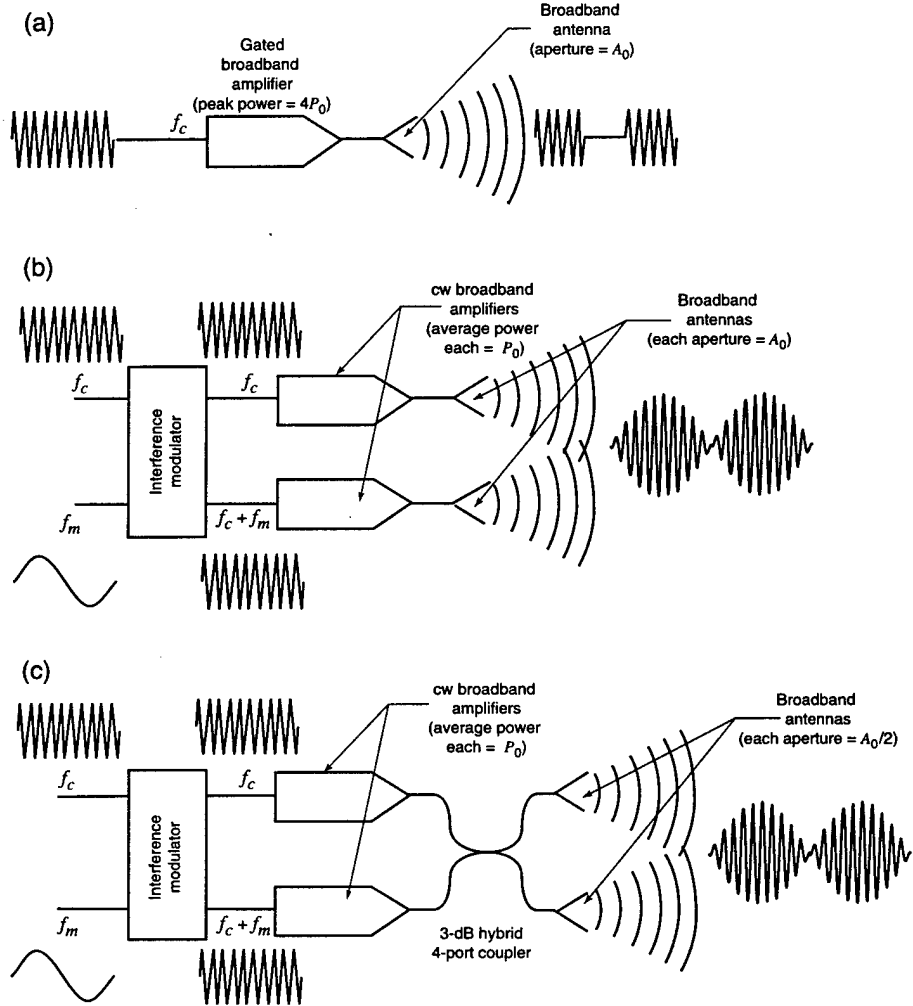
Figure 1: Conventional and IM-based schemes for producing a 50-percent duty, amplitude-modulated signal in far field: (a) conventional single antenna and pulsed amplifier ($A_e = A_0$); (b) two-channel IM system ($A_e = A_0$); (c) two-element phased-array IM system ($A_e = A_0/2$).
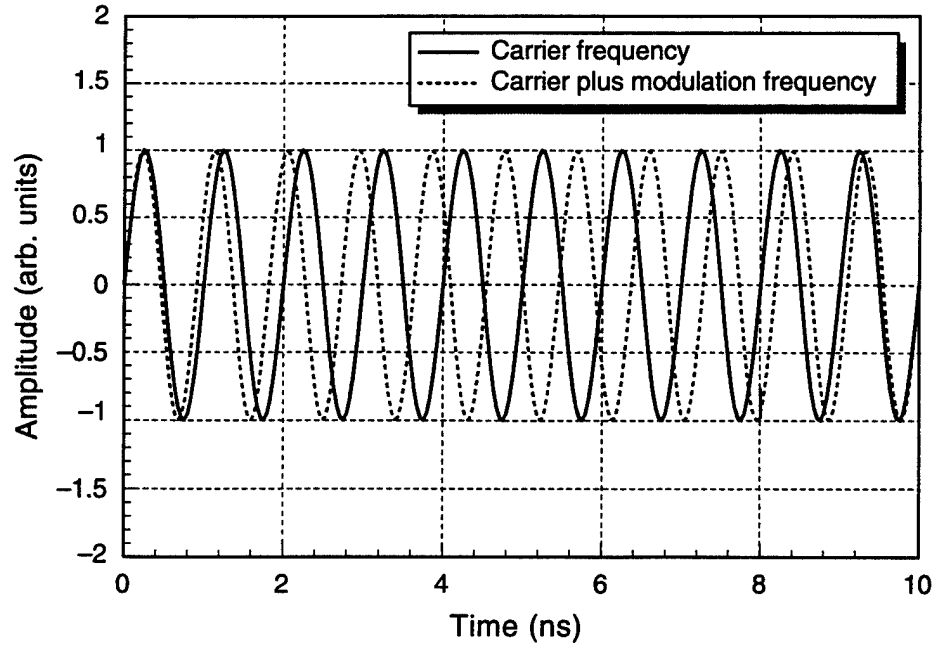
Figure 2: Sinusoidal waveforms ($f_0 = 1$ GHz, $f_1 = 1.1$ GHz).
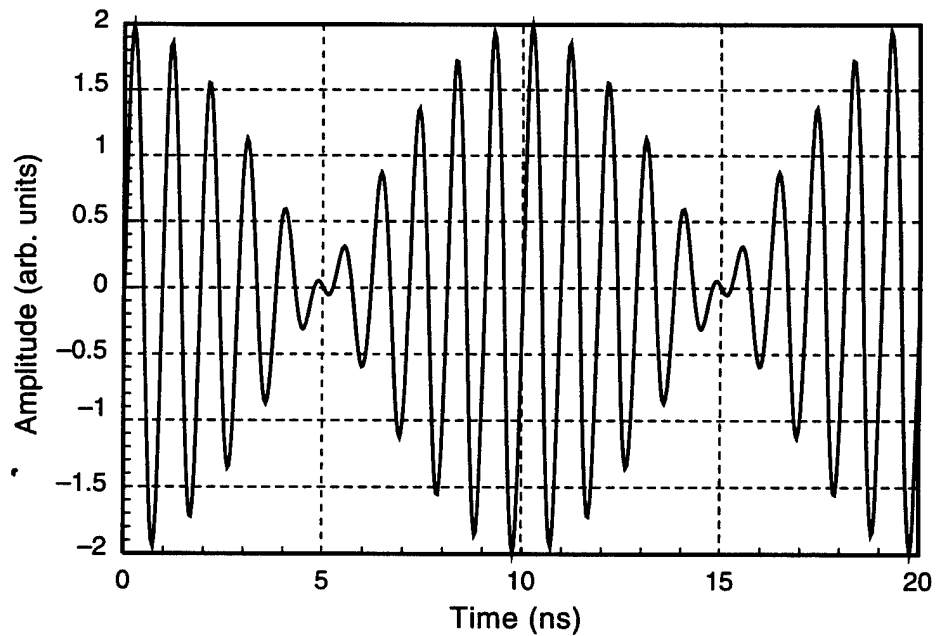
Figure 3: Sum of two sinusoidal waveforms in figure 2.
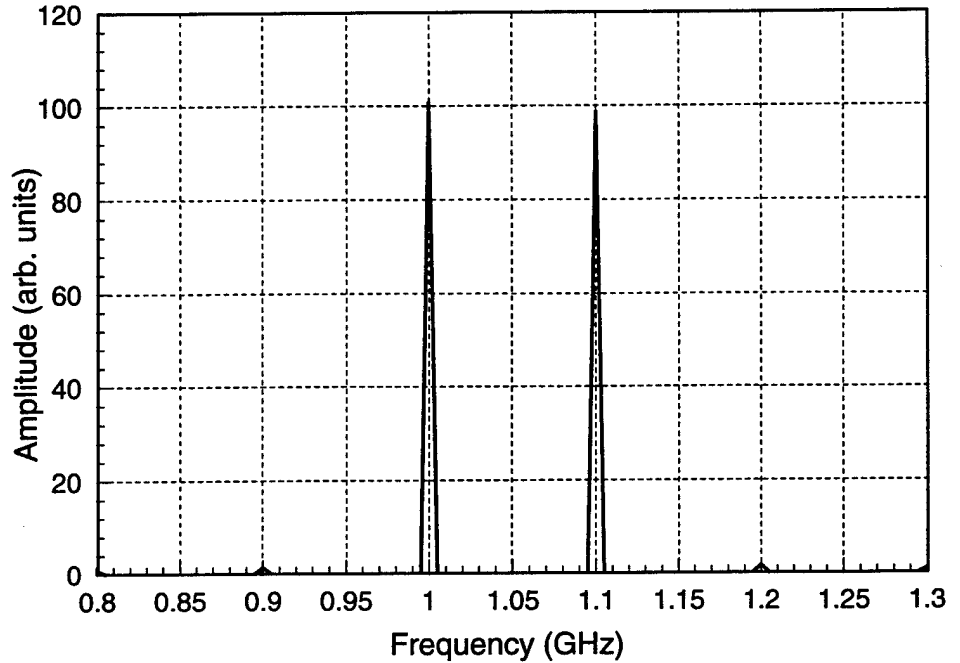
Figure 4: Fourier transform of beat wave of figure 3.

sion systems) as well as a reduced volume and weight compared to the equivalent single pulsed rf source (depicted in fig. 1(a)). Note that the equivalent antenna aperture, A_e, for each scenario in figure 1 has been normalized to the single antenna case (fig. 1(a)). Using the IM concept, we can produce the far-field wave interference effect either by a two-channel, non-phase-synchronized system requiring two antennas each of aperture A_0 as shown in figure 1(b), or by a two-element phased array requiring two antennas each with aperture $A_0/2$, as shown in figure 1(c).

In addition, the IM concept can be extended to n sources ($n \geq 2$); with proper phase control of the individual sources, more complicated rf waveforms and lower duty factors can be synthesized. Like the two-source IM, multiple-source IM systems have the advantages of lower voltage stress on the individual components and potentially lower system weight and volume.

7
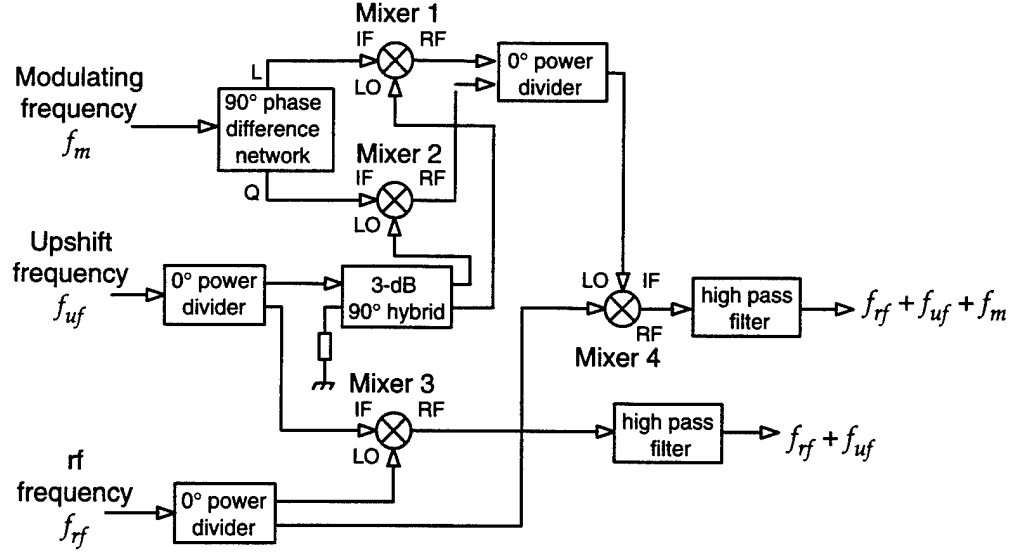
Figure 5: Block diagram of interference modulator.

2 Development of a Prototype Interference Modulator

A prototype IM was developed from a custom-designed active-network single-sideband modulator and commercially available microwave components (mixers, hybrids, and power combiner and dividers). A block diagram of the IM is shown in figure 5. The IM carrier frequency, f_c, is synthesized from two components — a constant upshift frequency, f_{uf}, and a tunable component, f_{rf}, such that $f_c = f_{uf} + f_{rf}$. The IM modulation frequency, f_{cm}, is synthesized from f_{rf} and a tunable low-frequency component, f_m, which has been frequency upshifted by f_{uf} so that $f_{cm} = (f_{rf} + f_{uf}) + f_m = f_c + f_m$. For the highest generation efficiency, $f_{uf} + f_m$ is produced by a single-sideband network composed of a broadband 90° phase-difference network with a power combiner and two double-balanced mixers to perform the frequency upshift.

2.1 Single-Sideband Generation Using I & Q Modulated Signals

2.1.1 Basic Approach

An I & Q modulation network takes a single sinusoidal waveform as input and generates two signals as output: an in-phase signal (I) and a quadrature signal (Q) that is 90° out of phase with the input signal. Such a network is also referred to as a 90° phase-difference network. The I and Q outputs can be further combined with microwave mixers, hybrids, and combiners to generate a single-sideband signal.

8
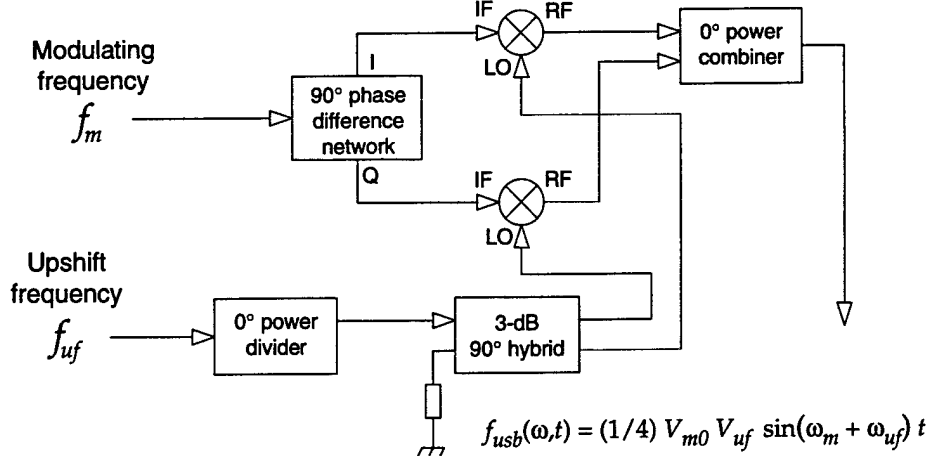
Figure 6: Block diagram of IM single-sideband modulation subsystem.

Figure 6 is a block diagram of the single-sideband modulation sub-network of the IM. The input waveform to the 90° phase-difference network has the form

$$f_m(\omega, t) = V_m \ \sin(\omega_m t), \tag{1}$$

where V_m is the voltage amplitude in volts, ω_m is the modulation frequency in radians per second, and t is the time in seconds.

After processing by the 90° phase-difference network, the I and Q output signals have the form

$$f_{mI}(\omega, t) = V_{mI} \ \sin(\omega_m t), \tag{2}$$

$$f_{mQ}(\omega, t) = V_{mQ} \ \sin(\omega_m t + \phi_m), \tag{3}$$

where V_{mI} and V_{mQ} are the voltage amplitudes of the I and Q signals, respectively, and ϕ_m is the phase difference introduced by the network. Ideally, $\phi_m = 90°$, but we treat it as a variable in order to perform an error sensitivity analysis.

The I and Q signals are then mixed with a sinusoidal upshift signal, f_{uf}, as shown in figure 6. The input upshift signal is described by

$$f_{uf}(\omega, t) = V_{uf} \ \sin(\omega_{uf} t), \tag{4}$$

where V_{uf} is the voltage amplitude of the upshift signal and ω_{uf} is the upshift frequency in radians per second.

9

After passing through a 0° power divider and a 3-dB 90° hybrid, the upshift signal is split into in-phase and quadrature signals:

$$f_{uf,I} = \frac{V_{uf}}{4}\ \sin(\omega_{uf}t), \tag{5}$$

$$f_{uf,Q} = \frac{V_{uf}}{4}\ \sin(\omega_{uf}t + \frac{\pi}{2}). \tag{6}$$

As shown in figure 6, the waveforms of equations (2) and (5) are combined in Mixer 1, and the waveforms of equations (3) and (6) are combined in Mixer 2. Using the trigonometric identities,

$$\sin A\ \sin B\ =\ \frac{1}{2}\left[\cos(A-B) - \cos(A+B)\right], \tag{7}$$

$$\cos(A \pm \frac{\pi}{2})\ =\ \mp\sin A, \tag{8}$$

we obtain the rf outputs of Mixers 1 and 2, respectively,

$$
\begin{aligned}
f_{\text{mix1}}(\omega, t)\ &=\ V_{mI}\sin(\omega_m t)\left(\frac{V_{uf}}{4}\right)\sin(\omega_{uf}t + \frac{\pi}{2}) \tag{9}\\
&=\ \frac{1}{8}\,V_{mI}V_{uf}\left\{\cos\left[(\omega_m - \omega_{uf})t - \frac{\pi}{2}\right]\right\} +\\
&\quad\ \frac{1}{8}\,V_{mI}V_{uf}\left\{\cos\left[(\omega_m + \omega_{uf})t + \frac{\pi}{2}\right]\right\}\\
&=\ \frac{1}{8}\,V_{mI}V_{uf}\left[\sin(\omega_m - \omega_{uf})t + \sin(\omega_m + \omega_{uf})t\right].
\end{aligned}
$$

$$
\begin{aligned}
f_{\text{mix2}}(\omega, t)\ &=\ V_{mQ}\sin(\omega_m t + \phi_m)\left(\frac{V_{uf}}{4}\right)\sin(\omega_{uf}t + \frac{\pi}{2}) \tag{10}\\
&=\ \frac{1}{8}\,V_{mQ}V_{uf}\left\{\cos\left[(\omega_m - \omega_{uf})t - \phi_m\right]\right\} +\\
&\quad\ \frac{1}{8}\,V_{mQ}V_{uf}\left\{\cos\left[(\omega_m + \omega_{uf})t + \phi_m\right]\right\}.
\end{aligned}
$$

If $\phi_m = \pi/2$ in equation (10), then equation (10) reduces to

$$
\begin{aligned}
f_{\text{mix2}}(\phi_m = \pi/2)\ =\ &\frac{1}{8}\,V_{mQ}V_{uf}\left[-\sin(\omega_m - \omega_{uf})t\right] + \tag{11}\\
&\frac{1}{8}\,V_{mQ}V_{uf}\left[\sin(\omega_m + \omega_{uf})t\right].
\end{aligned}
$$

Combining equations (9) and (11) using the 0° power combiner shown in figure 6, we obtain

$$f_{\text{sum}}(\omega, t) = \frac{1}{8} \left[(V_{mI} - V_{mQ}) \sin(\omega_m - \omega_{uf})t \right] + \tag{12}$$

$$\frac{1}{8} \left[(V_{mI} + V_{mQ}) \sin(\omega_m + \omega_{uf})t \right].$$

If $V_{mI} = V_{mQ} \equiv V_{m0}$, we obtain the upper sideband signal, f_{usb},

$$f_{\text{usb}}(\omega, t) = \frac{1}{4} V_{m0} V_{uf} \sin(\omega_m + \omega_{uf})t. \tag{13}$$

Note that if the LO inputs to Mixers 1 and 2 are exchanged (fig. 6), the output of the 0° power combiner will produce the lower sideband signal, f_{lsb}:

$$f_{\text{lsb}}(\omega, t) = \frac{1}{4} V_{m0} V_{uf} \sin(\omega_m - \omega_{uf})t. \tag{14}$$

2.1.2 Effect of Amplitude and Phase Error on Single-Sideband Generation

If the I and Q signals produced by the 90° phase-difference network are not of equal amplitudes or if the phase difference deviates from 90°, then when the I and Q signals are summed in the 0° power combiner (fig. 6), the unwanted (upper or lower) sideband will not be completely suppressed. To evaluate the effect of amplitude and phase error on single-sideband generation, we first sum equations (9) and (10):

$$
\begin{aligned}
f_{\text{mix1}} + f_{\text{mix2}} = \frac{1}{8} V_{uf} \{ &[V_{mI} - V_{mQ} \sin \phi_m] \sin(\omega_m - \omega_{uf})t \quad (15)\\
+ &[V_{mI} + V_{mQ} \sin \phi_m] \sin(\omega_m + \omega_{uf})t \\
+ &[1 - \cos(\omega_m + \omega_{uf})t] \cos \phi_m \cos(\omega_m - \omega_{uf})t \}.
\end{aligned}
$$

If the I and Q phase difference, ϕ_m, is close to 90°, $\cos \phi_m \simeq 0$ and the last line of equation (15) can be neglected. The remaining two lines of equation (15) represent the lower (LSB) and upper (USB) sideband terms, respectively. Taking the ratio of their amplitudes yields the relative suppression (enhancement) of one sideband over the other. From equation (15),

$$\frac{\text{LSB}}{\text{USB}} = 20 \log \left[\frac{V_{mI} - V_{mQ} \sin \phi_m}{V_{mI} + V_{mQ} \sin \phi_m} \right]. \tag{16}$$

11
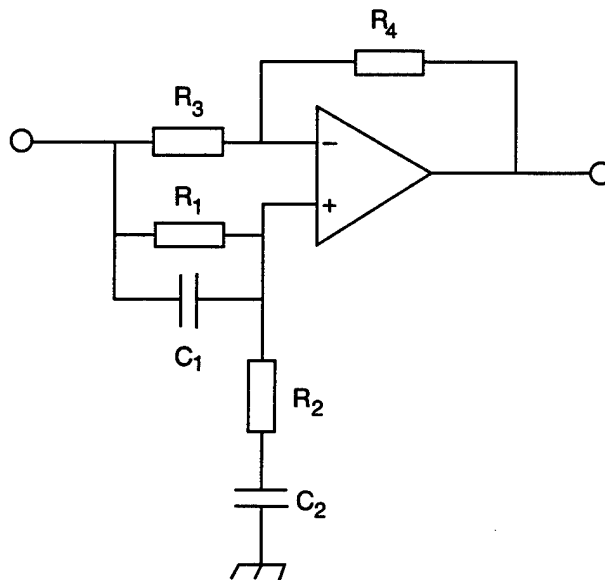
Figure 7: Basic all-pass filter schematic.

As an example, let $V_{mQ} = 0.8V_{mI}$ and $\phi_m = 88°$. Then the suppression of the lower sideband in decibels would be

$$\frac{\text{LSB}}{\text{USB}} = -19.06 \text{ dB}. \tag{17}$$

Thus, a 20-percent difference in the I and Q voltage amplitudes and a <3-percent phase error combine to produce a nonideal sideband suppression of -19.06 dB.

2.2 Synthesis of a Broadband 90° Phase-Difference Network

One of the unique components of the IM design is a broadband 90° phase-difference network capable of operating over a frequency range of four decades. The basic building block of the circuit is an active all-pass filter network, depicted in figure 7. Using the synthesis method described by Sebol,[1] we can use all-pass filter blocks to realize an even-order, broadband, active network that takes a single frequency as input and produces in-phase (I) and quadrature (Q, i.e., 90° out of phase) signals as outputs.

The synthesis of the 90° phase-difference network begins with the specification of the frequency range over which the network is to operate

---

[1]Ronald Sebol, "Design of Active 90-Degree Phase Difference Networks," Harry Diamond Laboratories, HDL-TM-68-18 (May 1968).

and a maximum phase error tolerance, δ, in radians. The frequency range is bounded by a starting frequency, f_a, and a cutoff frequency, f_b. From Bedrosian,[2] the design equations for the filter network are given by

$$k' = \frac{f_a}{f_b}, \tag{18}$$

$$k = \sqrt{1 - k'^2}, \tag{19}$$

$$K = K(k), \tag{20}$$

$$K' = K(k'), \tag{21}$$

$$\delta = 4\left(e^{-\pi \frac{K'}{K}}\right)^n. \tag{22}$$

where K is the complete elliptic integral of the first kind and n is the order of the filter.

From the above equations, we can determine the required number of filter sections (poles or zeros), n, for a given frequency band and phase ripple. Once the number of poles (zeros) is determined, the location of the jth pole (zero) is calculated from the elliptic functions cn and sn with the formula

$$p_j = \left(\frac{f_b}{f_a}\right)^{\frac{1}{2}} \frac{\text{cn}(u_j, k)}{\text{sn}(u_j, k)}, \tag{23}$$

where

$$u_j = \frac{4j + 1}{2n}K, \quad j = 0, 1, 2, \ldots, n - 1. \tag{24}$$

Using the relationships between the elliptic functions and the theta function, we can express equation (23) in a series expansion[3]

---

[2]S. D. Bedrosian, "Normalized Design of 90° Phase-Difference Networks," IRE Trans. Circuit Theory (June 1960), 128–136.

[3]Milton Abramowitz and Irene Stegun, eds., *Handbook of Mathematical Functions*, New York: Dover Publications, Inc., 1972, p. 575.

$$p_j = \frac{\displaystyle\sum_{l=0}^{\infty} q^{l(l+1)} \cos\left(2l+1\right) r_j}{\displaystyle\sum_{l=0}^{\infty} (-1)^l\, q^{l(l+1)} \sin\left(2l+1\right) r_j}, \qquad (25)$$

where

$$r_j = (-1)^j\, (2j+1)\, \frac{\pi}{4n}, \quad j = 0,1,2,\ldots,n-1, \qquad (26)$$

$$q = e^{-\pi\frac{K'}{K}}. \qquad (27)$$

A C++ program was written to evaluate equation (25) and to calculate the poles (zeros) and response (phase error) of the filter as a function of f_a, f_b, and n. Appendix A contains a listing of this program.

Once the network order has been determined by the phase error tolerance (maximum allowable deviation from 90°) and the bandwidth ratio, f_b/f_a, and the pole (zero) pairs have been generated by the C++ program, the component values for each stage of the filter network can be calculated. Referring to the nomenclature of figure 7 and using Sebol's synthesis equations,[1] we compute the component values as

$$R_2 = \frac{(p_a + p_b) \pm \sqrt{p_a^2 - 6p_a p_b + p_b^2}}{2p_a p_b}, \qquad (28)$$

$$R_1 = \frac{1}{p_a p_b R_2}, \qquad (29)$$

$$K = 3 + \frac{2R_2}{R_1}, \qquad (30)$$

$$R_4 = K R_3, \qquad (31)$$

where p_a and p_b are pole pairs computed from equation (25). If we choose $R_3 = 1$ kΩ, then equations (28) to (31) are sufficient to synthesize the filter. Note that the poles must be paired so that the discriminant of equation (28) is always greater than or equal to zero. Appendix B contains a listing of a C program that will automatically generate the network component values.

14

Table 1: Design specifications for 90° phase-difference network.

Specification	Value
Frequency range	100 Hz $\leq f_m \leq$ 1 MHz
Phase-error tolerance	$\pm 1°$
Carrier/sideband rejection	$\geq$ 25 dB

3 Design Example

As a design exercise, we developed an IM with a carrier frequency that was continuously tunable over a frequency range of approximately 800 MHz to 2 GHz and a modulation frequency continuously tunable from 100 Hz to 1 MHz. Table 1 summarizes the specifications for the 90° phase-difference network.

From table 1, the bandwidth ratio, defined as the ratio of the upper to the lower frequency, is given by $f_b/f_a = 10,000$. Using equation (22) and the code PHSESHFT.CC (app A), we found the network order to be $n = 12$. Assuming a phase error tolerance of $\pm 1°$, we calculated the roots from equation (25) (using the code PHSESHFT.CC).

The optimum pole pair combinations for the active network were determined with the code IQ_SYNTH.C (app B), which also generated the component values for each operational amplifier stage.

3.1 Design and Fabrication of 90° Phase-Difference Network

As synthesized by the method described in section 2.2, the 90° phase-difference network used six high-gain/bandwidth-product operational amplifiers to realize the in-phase and quadrature outputs over the required bandwidth. Several SPICE circuit models of the 90° phase-difference network were developed with both ideal and manufacturer-supplied operational amplifier models. We modified the ideal (computed) component values in the SPICE models to correspond with the values of commercially available components to ensure that the effect of the nonideal components on circuit performance was negligible. The schematic for the network is shown in figure 8. Based on the results of the SPICE modeling, a printed circuit board was fabricated, populated with components, debugged, and tested.
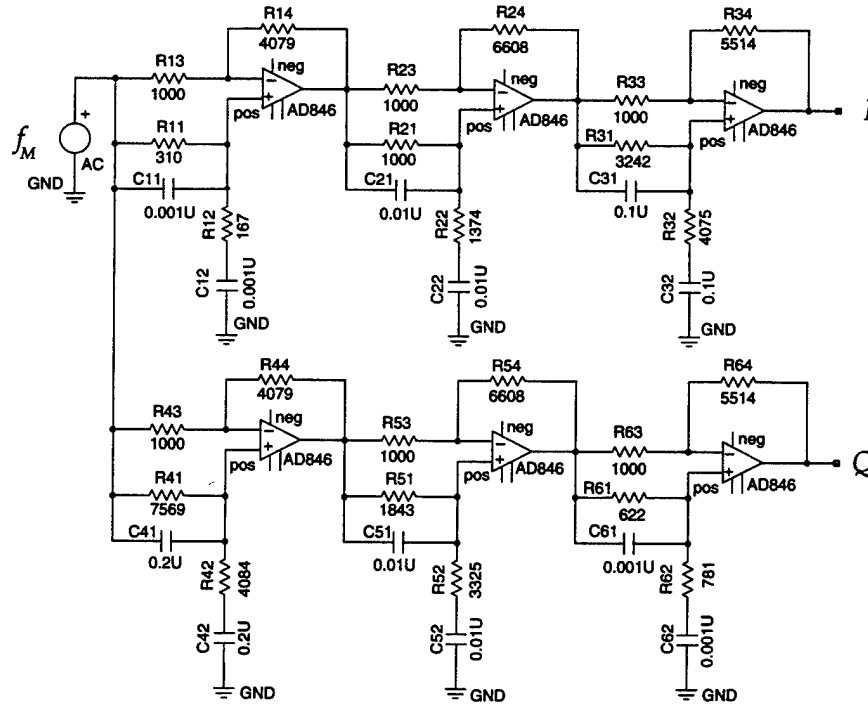
Figure 8: 90° phase difference network schematic.

3.1.1 Printed Circuit Board Layout and Fabrication

The mask for a two-layer printed circuit board (PCB) was generated on a DEC workstation with the aid of two CAD/CAM programs. Powerview (Viewlogic) was used to generate a circuit schematic and netlist from the SPICE input file. The netlist contained a logical map of the circuit components and their associated pin connections. SCICARDS (Harris) used this information to generate the physical layout of the PCB, including all the component locations, trace routings, and vias.

The two-layer PCB measured approximately 15×23 cm (6×9 in.). The board design includes electrical test points at each operational-amplifier stage to facilitate debugging. Board power can be provided externally from a dc power supply or internally from a set of four 9-V batteries. Gold-plated, right-angle SMA connectors were mounted on the board to accept the modulation frequency as input and to provide access to the in-phase (I) and quadrature (Q) outputs of the network. A tunable resistive divider network consisting of a series 10-turn 200-Ω potentiometer and a shunt 51-Ω resistor was placed at both the I and the Q output nodes so that their respective amplitudes could be matched.

16

All the components used standard library shapes, except for the electrolytic bypass capacitors, the battery holders, and the right-angle SMA receptacles. These shapes were defined in the user library. The hole-to-pad ratio for the user-defined shapes was approximately two to one. This ratio provided a sufficient amount of contact area when the discrete components were soldered onto the board. Surface-mount resistors (1-percent tolerance, size 1206) and capacitors (10-percent tolerance, size 1206) were used except for the bypass capacitors. SPICE simulations indicated that the circuit performance was highly sensitive to the capacitor values: a 10-percent deviation in the ideal capacitance value could produce undesirable results. Therefore, we hand-selected each chip capacitor to ensure that it was as close as possible to its ideal value; in addition, the circuit board contained extra pads in parallel with each capacitor to allow for additional trimming if necessary.

Although SCICARDS automatically generated a physical layout from the netlist, we edited this layout extensively to ensure optimum trace widths, routing, and positioning of all components. The criteria used for optimizing the trace routing were to minimize electrical path lengths and vias, and to simplify the board layout so as to facilitate assembly and testing. The traces were kept a reasonable distance from any holes or solder points on the board so that components would not be soldered to an incorrect path or pin. Line segments were reduced and kept as straight and long as possible. Unnecessary vias were eliminated, and vias placed too close to components or nets were relocated. The default via width was set at 0.062 in. We widened the power supply and ground trace widths with respect to the other traces to reduce the resistance in the circuit (0.004 in. as compared to 0.002 in. for non-supply or ground circuit traces). Test points, designed as a small hole within a large pad, were included at the input and output of the each amplifier stage.

The board used six very-high-speed operational amplifiers (Analog Devices AD846, small signal bandwidth = 46 MHz). Each operational amplifier used two bypass capacitors (a 2.2-μF electrolytic and a ceramic 0.1 μF) in parallel across the plus and minus voltage terminals to filter out unwanted frequencies generated by the power supply. If left unfiltered, these frequencies could lead to instabilities in the operation of the circuit. The bypass capacitors were placed as close as possible to the operational amplifiers to minimize stray inductances.

17
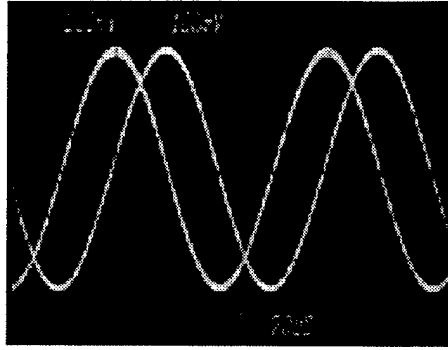
Figure 9: Typical oscilloscope trace of I and Q outputs, demonstrating 90° phase shift.

Once fully populated, the PCB was mounted in a standard 19-in. rack mount chassis along with the microwave mixers, hybrids, and combiners that completed the IM system. A 47-μF electrolytic capacitor and a 0.1-μF ceramic capacitor were used to bypass the external power supply, and a 0.4-μF capacitor was used to bypass the battery power supply mounted on the board.

3.1.2 90° Phase-Difference Circuit: Measured Performance

When first fabricated, the 90° phase-difference board exhibited high-frequency oscillations in both the I and Q output signals. We eliminated this problem by placing two small-valued capacitors across the inverting and non-inverting inputs of the third stage (I) op-amp and by placing one small-valued capacitor across the inverting and noninverting inputs of the sixth stage (Q) op-amp. We matched the output amplitudes of the I and Q signals using the potentiometers located at the two outputs. The output amplitudes of both branches remained matched and constant over the designed frequency range of 100 Hz to 1 MHz. Figure 9 is a typical time-domain oscilloscope trace of the I and Q outputs demonstrating the 90° phase shift.

The phase difference between the I and Q outputs as a function of frequency was measured with an HP 3575A gain-phase meter. The experimentally measured phase difference as a function of modulation frequency is plotted in figure 10, along with the SPICE predictions using both an ideal op-amp model and an Analog Devices AD846 op-amp model.

The corresponding measured and calculated phase error (i.e., devi-
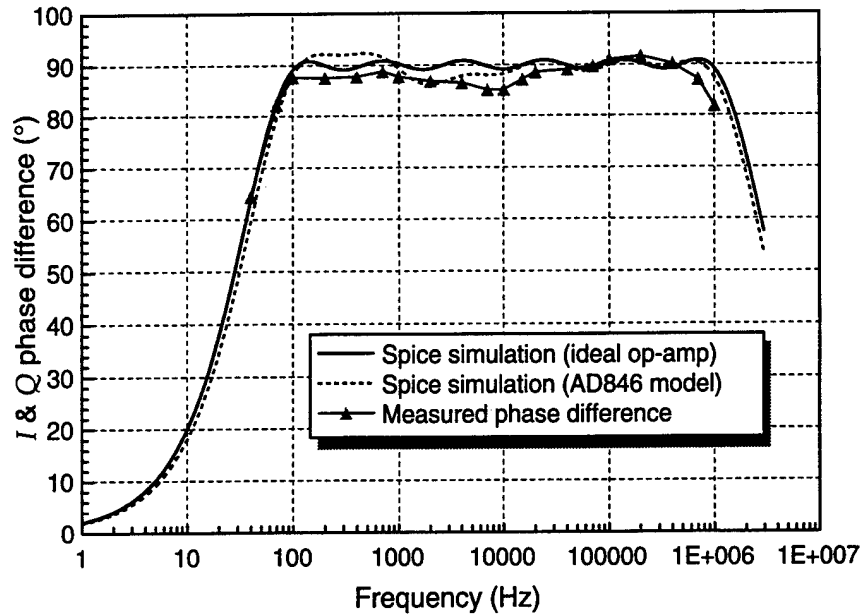
Figure 10: Calculated and measured phase difference between I and Q outputs of 90° phase-difference network.

ation from 90°) is plotted in figure 11. Using ideal components (op-amps with infinite gain-bandwidth products), the SPICE simulations show that the circuit synthesis method had achieved its design goals of $\pm 1°$ phase error tolerance over a frequency range of 100 Hz to 1 MHz. The predicted performance at the lower frequencies (<10 kHz) suffers when the more realistic, manufactured-supplied AD846 op-amp model is used. The measured phase error is poorer than either the ideal or nonideal simulations, probably because of stray capacitances and inductances that were not taken into account in the SPICE model. However, as discussed in section 3.2, the experimentally achieved phase error was more than sufficient for the performance of the prototype IM.

3.2 Single-Sideband Modulator Performance

After debugging, the 90° phase-difference network board was integrated into the single-sideband (SSB) modulator subsystem of the IM, as shown in the block diagram of figure 6. The modulating frequency, f_m, was provided by an HP651A oscillator, and the upshift frequency, f_{uf}, was provided by an HP8656A signal generator. Figure 12 is a typical plot of the spectral output of the SSB subsystem. As can be seen in the figure, the carrier and unwanted sideband frequencies are

19
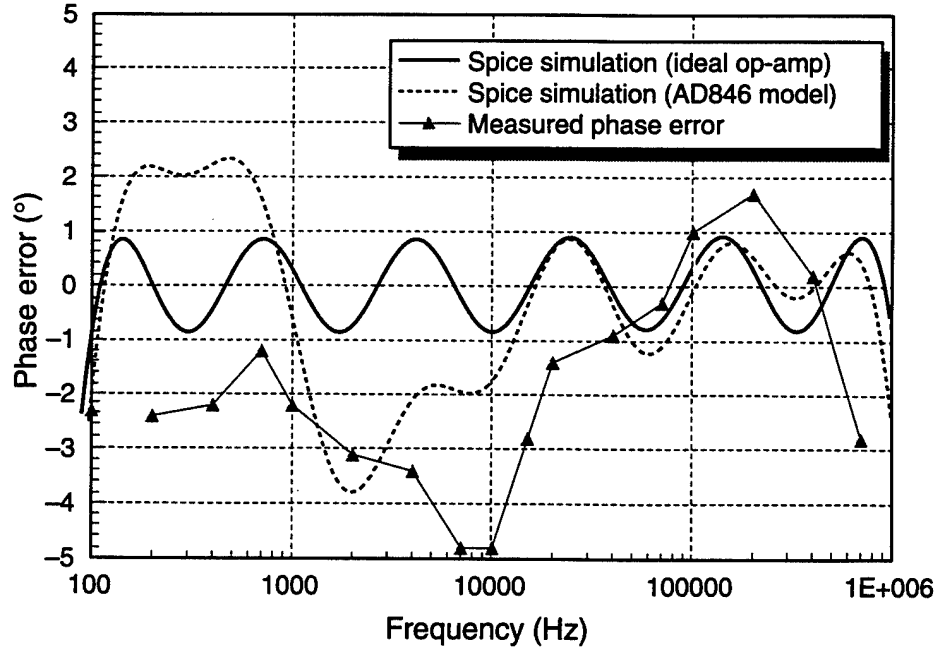
Figure 11: Calculated and measured phase error of the 90° phase-difference network.

suppressed at least −25 dB.

A key requirement for the SSB subsystem was operation over a very broad range of frequencies. The broadband behavior of the SSB subsystem is plotted in figure 13, where the measured suppression of the carrier and unwanted sideband is recorded as a function of modulating frequency.

As evidenced by the measured data, the single-sideband modulator design, as synthesized by the techniques of sections 2.1 and 2.2, is highly successful, producing a relatively pure tone with at least −25 dB of suppression of unwanted frequencies over a frequency span of four decades. Over the design frequency range of 100 Hz to 1 MHz, the measured average carrier suppression was −25.5 dB, and the measured unwanted sideband suppression ranged from −26 to −46 dB.

3.3 Measured Far-Field Performance of Interference Modulator

To evaluate the performance of the IM system, we post-amplified the IM carrier (f_c) and modulation ($f_c - f_m$) outputs and used them to drive two matched standard-gain horn antennas. Figure 14 is a schematic of the test setup. The initial outputs from the IM were amplified by

20
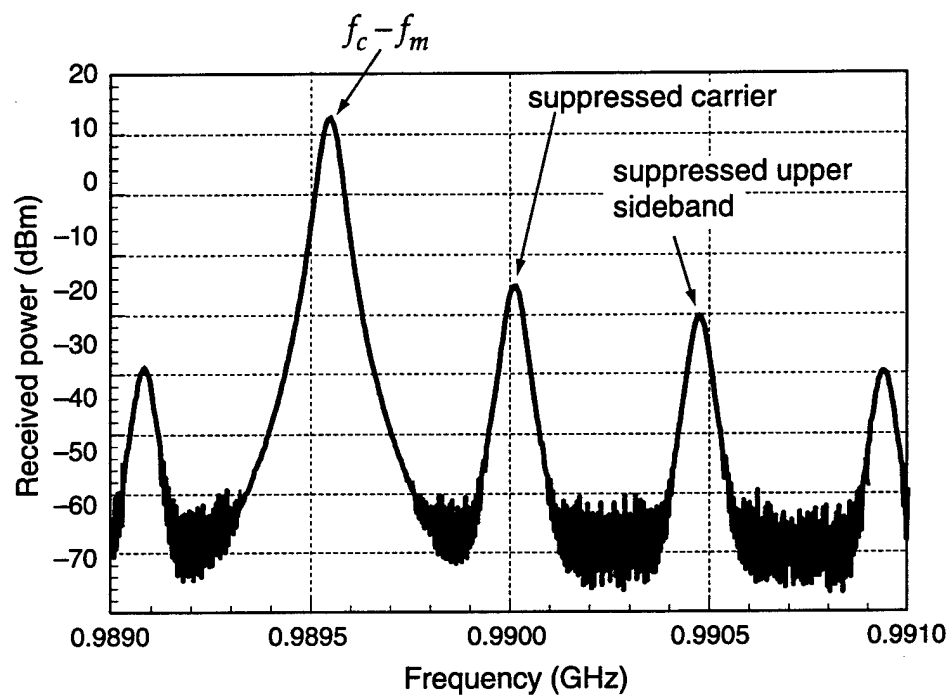
Figure 12: Typical measured spectrum of single-sideband modulator subsystem.
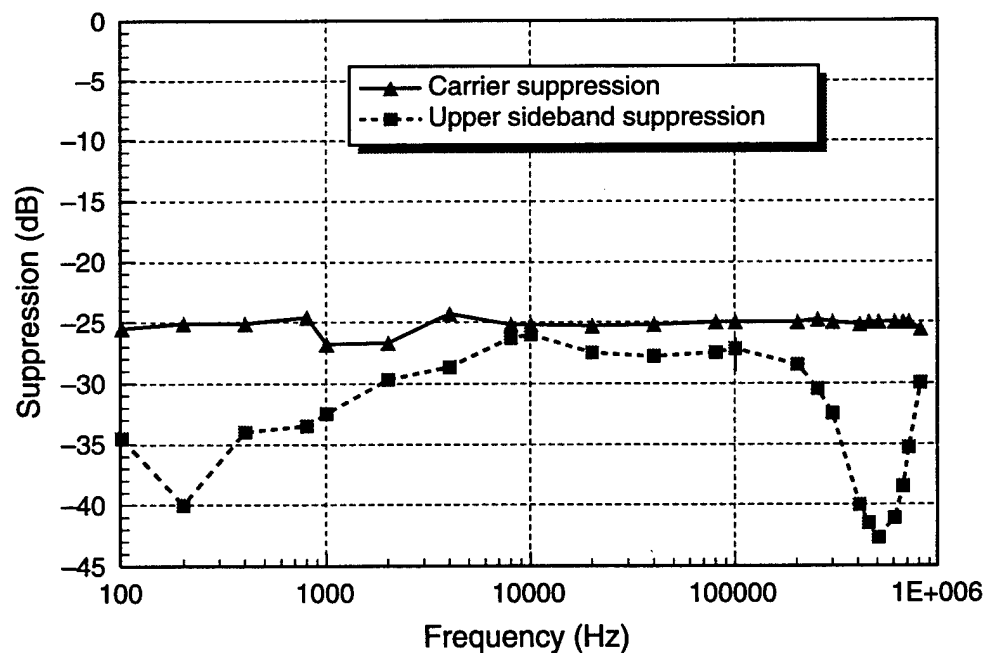
Figure 13: Carrier and unwanted sideband suppression.
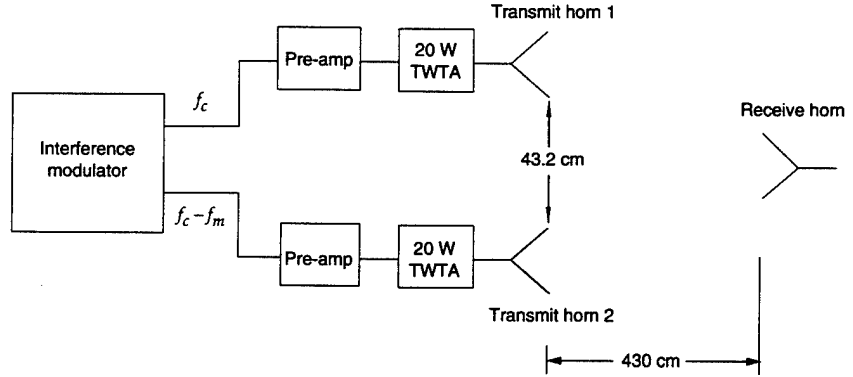
Figure 14: Block diagram of IM system test.

a pair of broadband solid-state amplifiers (Mini-Circuits ZFL-11AD, 8-dB gain). The outputs from the solid state amplifiers were then fed into a pair of Varian 20-W traveling-wave tube amplifiers, which in turn were connected to a pair of matched WR650 waveguide horns spaced 43.2 cm apart (edge-to-edge spacing). The horns radiated into a large anechoic chamber. A third WR650 horn was placed in the far field at a distance of 4.3 m from the transmitting horns to measure the output radiation.

Initially, the time- and frequency-domain output behavior of each of the transmitting horns was evaluated separately. With the IM set up to suppress the upper sideband, far-field measurements of the output of each of the transmitting horns were made in the anechoic chamber while the carrier and modulation frequencies were varied. Figures 15 to 18 are examples of far-field signals measured by the receiving horn with the system set up with the parameters summarized in table 2. The measured far-field time- and frequency-domain output from the carrier channel (transmit horn 1 in fig. 14) is shown in figures 15 and 16; the corresponding time- and frequency-domain output from the modulation channel (transmit horn 2 in fig. 14) is shown in figures 17 and 18. The spectral plots clearly show that the IM system can generate pure tones of frequencies f_c and $f_c - f_m$ and propagate them without distortion into free space.

To verify that the outputs of the two channels of the IM were matched over the full modulation frequency band, we swept the mod-

Table 2: Sample setup parameters for far-field IM measurements.

Parameter	Value
Carrier frequency, f_c	990 MHz
Modulation frequency, f_m	463 kHz
IM sideband suppressed	Upper sideband
Transmit horn spacing	43.2 cm
Receive horn range	430 cm
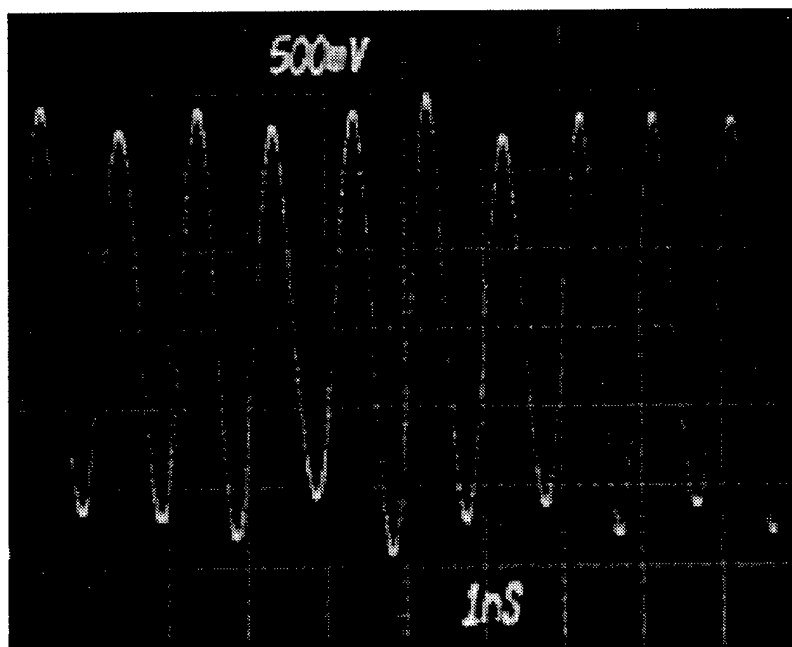
Figure 15: Measured far-field time-domain signal from IM carrier channel (transmit horn 1).

23
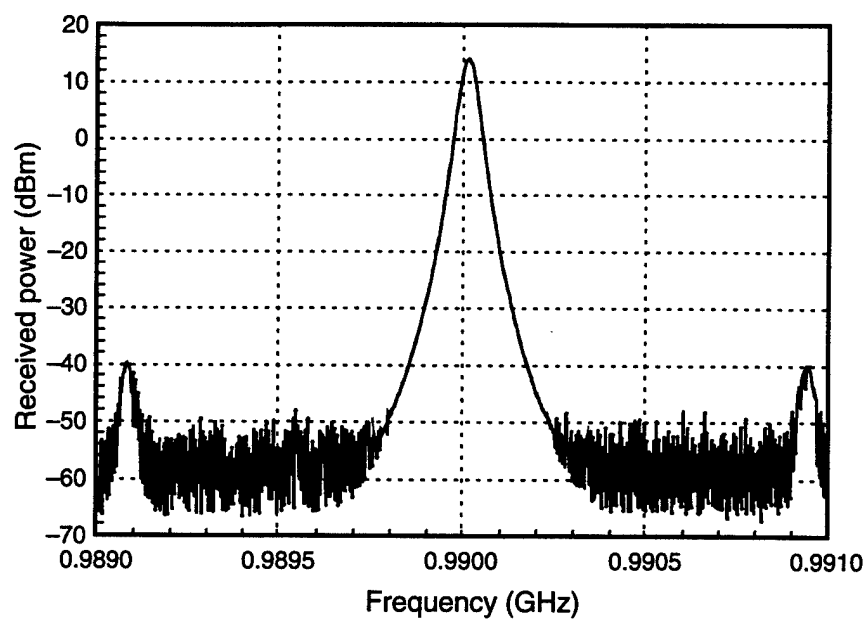
Figure 16: Measured far-field frequency-domain signal from IM carrier channel (transmit horn 1).
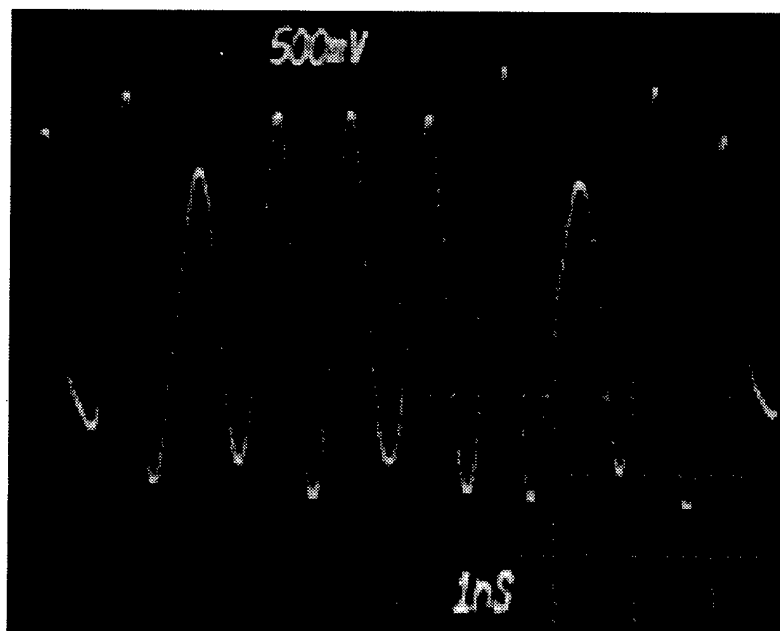
Figure 17: Measured far-field time-domain signal from IM modulation channel (transmit horn 2).
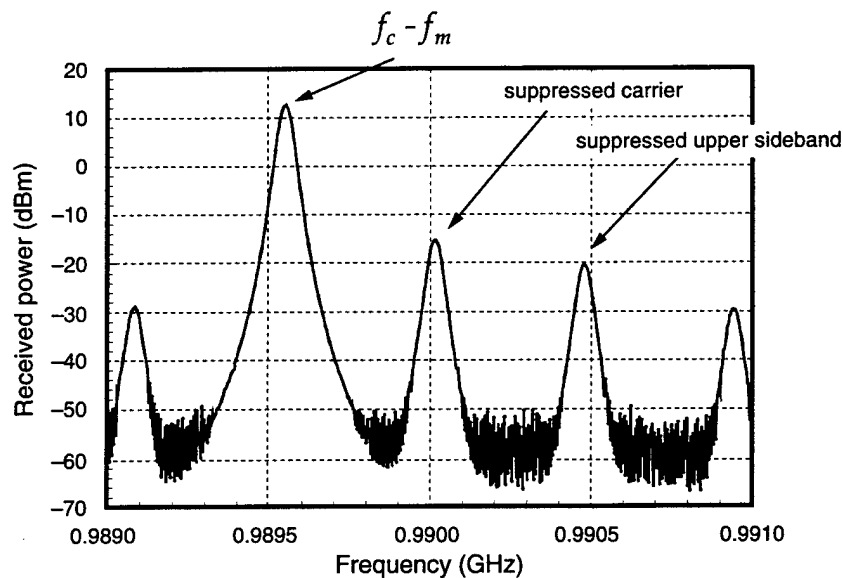
Figure 18: Measured far-field frequency-domain signal from IM modulation channel (transmit horn 2).

ulation input from 100 Hz to 1 MHz, and the far-field power radiated from each channel was recorded by the receiving horn. Figure 19 is a plot of the received power from each transmitting horn as a function of the modulating frequency. As seen in the figure, the amplitudes of the two output channels are well matched over the desired frequency range.

After each individual channel of the IM system was evaluated, the two transmitting horns were driven simultaneously (as in normal operation). With the carrier frequency fixed at $f_c = 990$ MHz, the modulating frequency was swept from 100 Hz to 1 MHz and the far-field time- and frequency-domain behavior of the IM system was recorded by the receiving antenna (see fig. 14). Figures 20 to 23 are plots of the measured time- and frequency-domain data for modulation frequencies of 463 kHz and 100 Hz. Note that for clarity, the abscissa of figure 23 plots the measured frequency, f, minus the carrier frequency, $f_c = 990$ MHz. As can be seen from the figures, the radiated waveforms from the two channels of the IM system linearly combine in the far field to produce a waveform with a duty factor of 50 percent, a carrier frequency, f_c, equal to the input carrier frequency, and a modulation envelope frequency equal to the difference between the input carrier and modulation frequencies $(f_c - f_m)$.

25
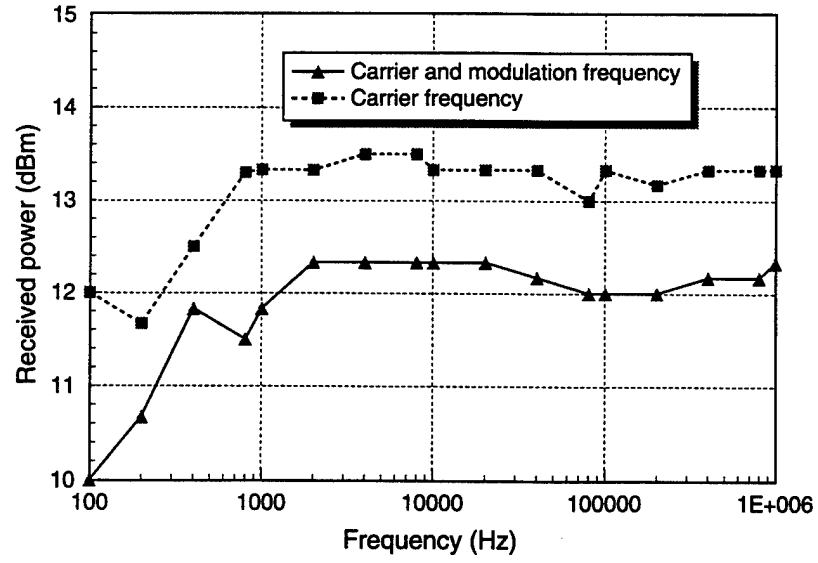
Figure 19: Received power as a function of modulating frequency, f_m, from each of two transmitting horns.
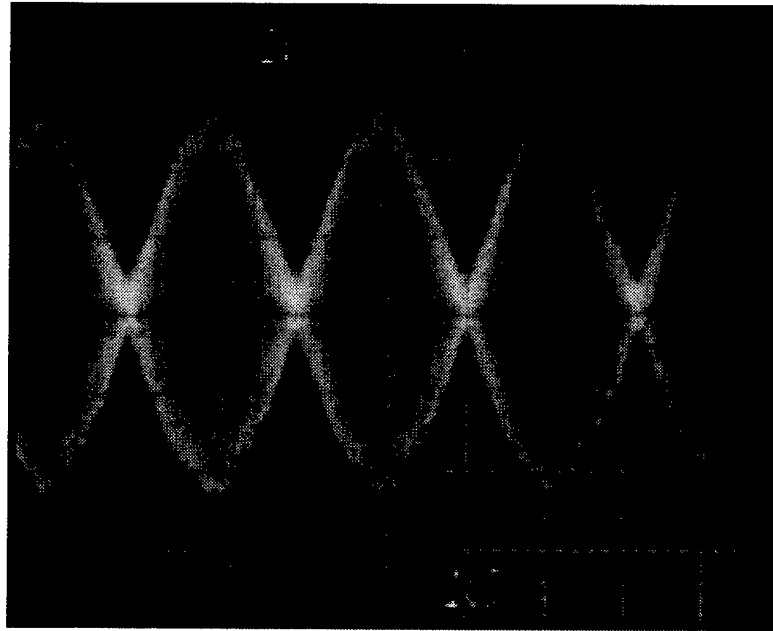
Figure 20: Measured far-field ($r = 430$ cm) time-domain behavior of IM for $f_c = 990$ MHz, $f_m = 463$ kHz.
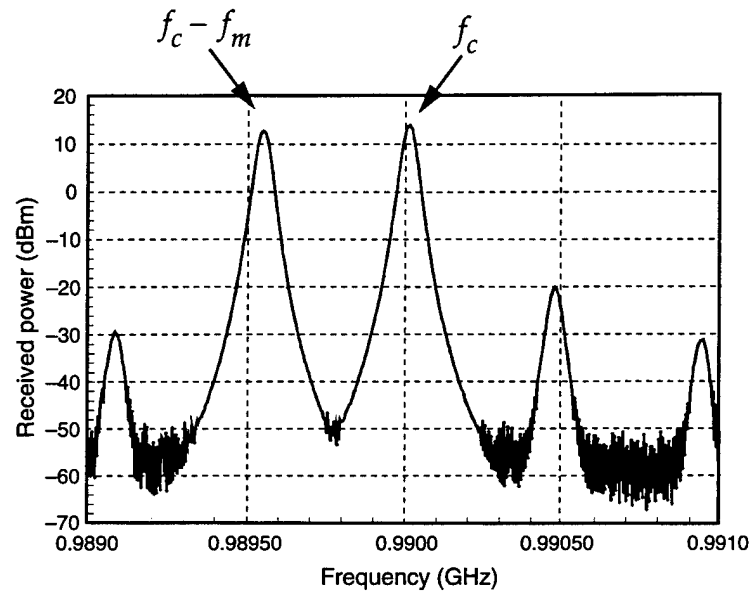
Figure 21: Measured far-field ($r = 430$ cm) frequency-domain behavior of IM for $f_c = 990$ MHz, $f_m = 463$ kHz.
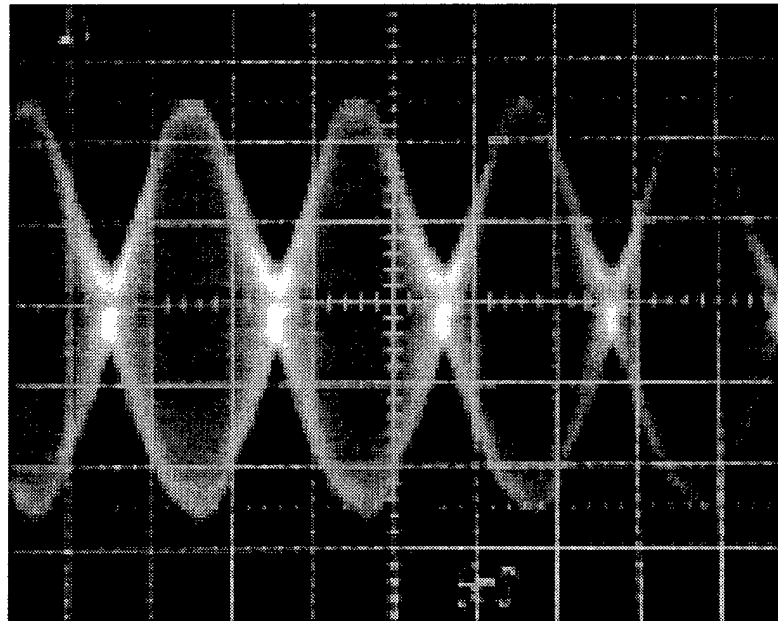
Figure 22: Measured far-field ($r = 430$ cm) time-domain behavior of IM for $f_c = 990$ MHz, $f_m = 100$ Hz.

27
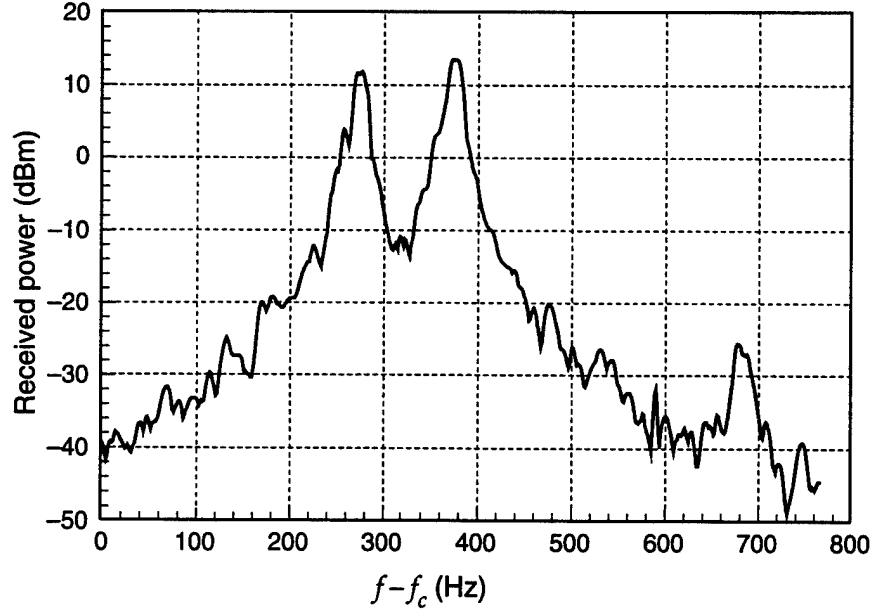
Figure 23: Measured far-field ($r = 430$ cm) frequency-domain behavior of IM for $f_c = 990$ MHz, $f_m = 100$ Hz.

4 Conclusions

The IM is a broadband microwave source with independently tunable carrier and modulation frequencies. A general methodology for the design of an n-channel ($n \geq 2$) IM has been developed, along with computer codes (see the appendices) to assist in the design process. To validate the design methodology, we developed a prototype two-channel device. We give details of the design and fabrication of key subsystems, such as a broadband 90° phase-difference network and single-sideband modulator. The prototype IM was experimentally shown to have met its design specifications for the carrier frequency ($0.8 \leq f_c \leq 2$ GHz) and modulation frequency (100 Hz $\leq f_m \leq 1$).

A Code Listing for Pole-Zero Solver

```
// phaseshift.cc

#define MAIN

#include "dblvec.h"
#include "dblply.h"
#include "cmplxdbl.h"
#include "graphic2d.h"

double ac[5] ={1.38629436112,0.09666344259,0.03590092383,
0.03742563713,0.01451196212};
double bc[5] =
{0.5,0.12498593597,0.06880248576,0.3328355346,0.00441787012};

DblPoly Ka(4,ac);
DblPoly Kb(4,bc);

double Kelliptic(double m);
double Elliptic_Ratio(double q,double r,double tol = 1.e-6,int
nmax = 20);
CmplxDbl Filter_Response(double w,DblVec poles);

// Complete Elliptic Integral of the 1st kind

double Kelliptic(double m)
{
    double m1 = 1.0-m;

    return(Ka(m1)+log(1.0/m1)*Kb(m1));
}

// Elliptic Ratio

double Elliptic_Ratio(double q,double r,double tol,int nmax)
{
    double num0,den0,rat0;
```

```
    double num = cos(r);
    double den = sin(r);
    double rat = num/den;

    double qi = 1.0;
    double q2 = q*q;
    double qrat = 1.0;

    for ( int i = 1; i < nmax; ++i)
    {
        qrat *= q2;
        qi *= qrat;

        rat0 = rat;
        num0 = num;
        den0 = den;

        num += qi*cos(((double)(2*i+1))*r);

        if ( i%2 != 0 )
        {
            den -= qi*sin(((double)(2*i+1))*r);
        }
        else
        {
            den += qi*sin(((double)(2*i+1))*r);
        }

        rat = num/den;

        if ( fabs(rat-rat0) < tol*fabs(rat) ) return(rat);
    }

    error("\n%d iterations not enough for %g
tolerance\n",nmax,tol);

    return(0.0);
```

```
}


// Calculate response of an all pass filter with a given set of poles

CmplxDbl Filter_Response(double w,DblVec poles)
{
    CmplxDbl Z(1.0,0.0);
    CmplxDbl z(0.0,w);

    for ( int i = 0; i < poles.dim(); ++i)
    {
        Z *= (z-poles(i))/(z+poles(i));
    }

    return(Z);
}

main(int argc,char **argv)
{
    FILE *outfile = NULL;

    if ( argv[2] ≠ NULL ) outfile = createascii(argv[2]);

    ArgLst *args = new ArgLst(argv[1]);

/*
    Inputs for 90 degree phase shift filter synthesis program


        npoles    = number of poles in filter (default 8)
        flo       = Minimum frequency (Hz) at which 90 degree
                    phase shift is required (default 100)
        fhi       = Maximum frequency (Hz) at which 90 degree
                    phase shift is required (default 1e6)
        nplot     = Number of frequency points to calculate filter
                    response at (default 101)
        flo_plot  = Minimum plot frequency (Hz) (default 10)
        fhi_plot  = Maximum plot frequency (Hz) (default 10e6)
```

31

```
*/

    Int *npoles_ = new Int("npoles",NULL,"8");
    args→addarg((Arg *)npoles_);


    Double *flo_ = new Double("flo",NULL,"100");
    args→addarg((Arg *)flo_);


    Double *fhi_ = new Double("fhi",NULL,"1000000");
    args→addarg((Arg *)fhi_);


    Int *nplot_ = new Int("nplot",NULL,"101");
    args→addarg((Arg *)nplot_);


    Float *flo_plot_ = new Float("flo_plot",NULL,"10");
    args→addarg((Arg *)flo_plot_);


    Float *fhi_plot_ = new Float("fhi_plot",NULL,"10000000");
    args→addarg((Arg *)fhi_plot_);


     args→parse();


    double flo = flo_→val();
    double fhi = fhi_→val();
    int npoles = npoles_→val();


    // Calculate pole and zero locations for filter parameters


    double Wrat = fhi/flo;


    double Wrat_inv = 1.0/Wrat;
    double Wrat_root = sqrt(Wrat);
    double m = 1.0-Wrat_inv*Wrat_inv;
    double k = sqrt(m);
    double kp = sqrt(1.0-m);
    double theta = asin(k);
    double m1 = 1.0-m;
    double K = Kelliptic(m);
```

```cpp
double Kp = Kelliptic(m1);
double q = exp(-pi*Kp/K);
double delta = 4.0*pow(q,(double)npoles);
double dB_Reject = 20*log10(1.0/tan(delta/2));

// Print locations of poles and zeros for circuit synthesis

printf("\n(fhi/flo) = %lg n = %d flo = %lg Hz fhi = %lg
Hz",Wrat,npoles,flo,fhi);

printf("\ndelta = %lg deg q = %0.8lg theta =
%0.8lg\n",delta/dgtord,q,
        theta/dgtord);

printf("\nSideband Rejection = %lg dB\n",dB_Reject);

DblVec pole(npoles);

for ( int ipole = 0; ipole < npoles; ++ipole)
{
    double r_ipole =
((double)(2*ipole+1))*pi/(double)(4*npoles);

    if ( ipole%2 != 0 ) r_ipole = -r_ipole;

    pole(ipole) = Elliptic_Ratio(q,r_ipole);

    printf("\nr%d = %0.6lg\t\t p%d =
%0.6lg",ipole,r_ipole/dgtord,ipole,
            pole(ipole));
}
cr();

float flo_plot = flo_plot_->val();
float fhi_plot = fhi_plot_->val();
int nplot = nplot_->val();

float log_fac = log(fhi_plot/flo_plot)/(float)nplot;
```

33

```
FltVec freq(nplot);

FltVec phase(nplot);
FltVec amp(nplot);

float fnorm = sqrt(flo*fhi);

for ( int iplot = 0; iplot < nplot; ++iplot)
{
    freq(iplot) = flo_plot*exp((float)iplot*log_fac);
    double omega = (double)(freq(iplot)/fnorm);

    CmplxDbl Z = Filter_Response(omega,pole);

    phase(iplot) = Z.phasedeg();
    amp(iplot) = Z.abs();
}

//printcols("                        Hz          deg",
NULL,stdout,1024,&freq,&phase,&amp,NULL);

// PLot response of filter

Graph *grph0 = Graph_init("X11","postscript",outfile);
Graph *grph_plt = new Graph(grph0,0.65,0.75,0.1,0.1);
Graph *grph_title = new Graph(grph0,0.65,0.1,0.1,0.9);
Graph *grph_poles = new Graph(grph0,0.2,0.75,0.77,0.1);

Axis *Xax = new Axis(0,0,0,"Hz");
Axis *Yax = new Axis(0,0,0,"degrees");

Xax→setlog();

XYplot *Phase_plot = new XYplot(grph_plt,Xax,Yax);

Phase_plot→setframe();
Phase_plot→setgrid(RED,NULL);
```

```
Curve2d *phase_crv = new Curve2d(&freq,&phase);

TextWin *title = new TextWin(grph_title);

title→addtext("Constant 90 deg %d pole Phase Shift
Filter",npoles);
title→addtext("flo = %lg Hz fhi = %lg Hz",flo,fhi);
title→addtext("delta = %lg deg Rejection = %lg
dB",delta/dgtord,dB_Reject);

TextWin *poles = new TextWin(grph_poles);

poles→setxalign(LEFT_ALIGN);
poles→addtext("Zeros");

double freq_scal = sqrt(flo*fhi);

for ( ipole = 0; ipole < pole.dim(); ++ipole)
{
    poles→addtext("%+0.6le",pole(ipole));
}

Phase_plot→addtext(title);
Phase_plot→addtext(poles);

Phase_plot→addcurve(phase_crv);

Phase_plot→draw();

Phase_plot→draw(PROMPT);

Graph_term();

for ( ipole = 0; ipole < pole.dim(); ipole += 2 )
{
    double p_plus = twopi*freq_scal*pole(ipole);
    double p_minus = twopi*freq_scal*pole(ipole+2);
```

```
        double p_avg = (p_plus+p_minus)/2.0;
        double p_del = p_plus-p_minus;

        printf("\np+ = %+06le p- = %+0.6le\n<p> = %+0.6le
dp = %+0.6le",
                p_plus,p_minus,p_avg,p_del);

        double D = sqrt((p_del*p_del-p_avg*p_avg)/3.0);

        double tau_plus = p_avg-D;
        double tau_minus = (p_avg+D)/2.0;

        tau_plus = 1.0/tau_plus;
        tau_minus = 1.0/tau_minus;

        double K = 2*tau_minus/tau_plus+3;

        printf("\ntau+ = %+0.6le tau- = %+0.6le K = =
%+0.6le",
                tau_plus,tau_minus,K);

    }

    cr();

    exit(0);

}
```

B   Code Listing for Component Value Generator

```c
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#define PI 3.1415926
#define TRUE 1
#define FALSE 0
#define DimArray1 7
#define MaxPoles 12

/**************************************************/

void pause()
{
    int c;
    printf("Press <return> to continue:   ");
    c=getchar();
}

/**************************************************/

double discrim(pa,pb)
double pa, pb;
{
    return((pa*pa)-(6*pa*pb)+(pb*pb));
}

/**************************************************/

void synthesize_filter(n,wc,T)
int n;
double wc;
double T[DimArray1];
{
```

```
int i,j,k,l,m,npair,iter;
int neg_flag,quit,swap;
char answer;
double temp,Q[DimArray1];
double pa,pb;
double r1nrm,r2nrm,K,Xc,C,temp2;


/*  iterate through the various combinations of pole pairs  */
/*  load the pole pair set into the array Q[i]              */


iter=1;

for(i=1;i≤(n-1);i++)
{
    Q[1]=T[1];       /*  first pole is always the same as T[1]  */
    Q[2]=T[i+1];

    for(j=3;j≤n;j++)
    {
        if(j≠(i+1))
            Q[j]=T[j];
        else
            Q[j]=T[2];
    }

    for(k=1;k≤(n-3);k++)
/*  shuffle the rest of the pole pairs  */
    {
        switch (k)
        {
         case 1:
           {
                break;
           }
         case 2:
           {
                temp=Q[4];
                Q[4]=Q[5];
```

```c
            Q[5]=temp;
            break;
          }
        case 3:
          {
            temp=Q[4];
            Q[4]=Q[6];
            Q[6]=temp;
            break;
          }
        default:
          {
            printf("HEY! I got nothin' to do!!!!\n");
            break;
          }
      }
      printf("\nIteration #%d:\n",iter);
      iter=iter+1;

/*   print out the particular pole pairs   */

      for(l=1;l≤(n-1);l++)
        printf(" %g", Q[l]);
      printf(" %g\n", Q[n]);

/*   test to see if any discriminants are negative   */

      neg_flag=FALSE;

      for(m=1;m≤(n-1);m=m+2)
      {
        pa=Q[m];
        pb=Q[m+1];
/*          printf("   discrim[%d] = %g\n",m,discrim(pa,pb));*/
        if(discrim(pa,pb)<0) neg_flag=TRUE;
      }

/*   if no discriminants are < 0, then synthesize filter   */
```

```c
        if(neg_flag≠TRUE)
        {
            printf("\n Passed discriminant test...\n");

            quit=FALSE;
            swap=FALSE;
            while(quit≠TRUE)
            {
                npair=1;
                for(m=1;m≤(n-1);m=m+2)
                {

/*   compute normalizing reactance, Xc   */

                    printf("\n Input C (uF) for pole pair #%d: ",npair);

                    scanf("%lf", &C);
                    Xc=1.0/(C*1.0e-6*wc);
                    printf(" Xc = %g\n", Xc);

/*   compute normalized R1 and R2   */

                    pa=Q[m];
                    pb=Q[m+1];

                    r2nrm=fabs(((pa+pb) +
sqrt(discrim(pa,pb)))/(2*pa*pb));
                    r1nrm=fabs(1.0/(pa*pb*r2nrm));

/*   if swap=TRUE, do R1 <- R2/2 and R2 <-
 2R1   to try to get   */
/*   more favorable K value                          */

                    if(swap==TRUE)
                    {
                        temp2=r2nrm;
                        r2nrm=2*r1nrm;
```

40

```c
                        r1nrm=temp2/2;
                    }

                    K=3+(2*r2nrm/r1nrm);

/*                      printf("\n    R1nrm = %g
R2nrm = %g\n",r1nrm,r2nrm);*/
                    printf("\n R1 = %g R2 = %g K = %g\n",
                        (r1nrm*Xc),(r2nrm*Xc),K);
                    npair=npair+1;
                }

                printf("\n Try again <y/n>?   ");
                scanf("%s",&answer);
                if((answer=='y')||(answer=='Y'))
                {
                    swap=FALSE;
                    printf("\n Swap R1 and R2 <y/n>?   ");
                    scanf("%s", &answer);
                    if((answer=='y')||(answer=='Y'))
                        swap=TRUE;
                }
                else quit=TRUE;
            }
        }
        else printf("\n Failed discriminant test (discrim <
0)...\n");
        }
    }
}

/**************************************************/

main()
{
    FILE *infile;
    int i,j,lim,npoles;
    double temp,P[MaxPoles+1],S[7];
```

41

```
      double fa, fb, wa, wb, wc;

/*   get total number of poles and pole values from input file */

      infile=fopen("input.dat", "r");
      fscanf(infile, "%d", &npoles);

      for(i=1;i≤npoles;i++)
          fscanf(infile, "%lf", &P[i]);

/*   get frequency limits  */

      fscanf(infile, "%lf%lf", &fa, &fb);
      wa=2*PI*fa;
      wb=2*PI*fb;
      wc=sqrt(wa*wb);

      fclose(infile);

/*   order poles from most positive to least positive  */
/*   and store in P[i]                                  */

      i=1;
      while(i≤npoles)
      {
          for(j=(i+1);j≤npoles;j++)
          {
              if(P[i] < P[j])
              {
                  temp=P[i];
                  P[i]=P[j];
                  P[j]=temp;
              }
          }
          i=i+1;
      }

/*   load in positive valued poles  */
```

```
        for(i=1;i≤(npoles/2);i++)
            S[i]=P[i];

        printf("\n PROCESSING POSITIVE VALUED ROOTS (I)\n");
        printf("
====================================");
        synthesize_filter((npoles/2),wc,S);

/*   load in negative valued poles */

        for(i=1;i≤(npoles/2);i++)
            S[i]=P[i+(npoles/2)];

        printf("\n PROCESSING NEGATIVE VALUED ROOTS (Q)\n");
        printf("
====================================");
        synthesize_filter((npoles/2),wc,S);

}
```

C SPICE Input Files

C.1 90° Phase-Difference Network – Ideal Op-Amp Model

```
*******************************
*
* 12 pole I & Q all-pass network
* fa=100 Hz fb=1 MHz DKA 3/8/94
* ideal op amp model
*
*******************************
vin 1 0 0.0 ac 1.0 0.0
*******************************
* op amp 1 (I ckt)
*******************************
xamp1 3   2   5 opamp
r11    1   3   310.0
c11    1   3   0.001e-6
r12    3   4   167.0
c12    4   0   0.001e-6
r13    1   2   1000.0
r14    2   5   4079.0
*******************************
* op amp 2 (I ckt)
*******************************
xamp2 7   6    9 opamp
r21    5   7   762.0
c21    5   7   0.01e-6
r22    7   8   1374.0
c22    8   0   0.01e-6
r23    5   6   1000.0
r24    6   9   6608.0
*******************************
* op amp 3 (I ckt)
*******************************
xamp3 11 10   13 opamp
r31    9  11 3242.0
c31    9  11 0.1e-6
r32    11 12 4075.0
```

```
c32    12  0 0.1e-6
r33     9  10 1000.0
r34     10 13 5514.0
rl3     13 0  50.0
*****************************
* op amp 4 (Q ckt)
*****************************
xamp4 23 22   25 opamp
r41     1  23 7569.0
c41     1  23 0.2e-6
r42     23 24 4084.0
c42     24 0  0.2e-6
r43     1  22 1000.0
r44     22 25 4079.0
*****************************
* op amp 5 (Q ckt)
*****************************
xamp5 27 26   29 opamp
r51     25 27 1843.0
c51     25 27 0.01e-6
r52     27 28 3325.0
c52     28 0  0.01e-6
r53     25 26 1000.0
r54     26 29 6608.0
*****************************
* op amp 6 (Q ckt)
*****************************
xamp6 31 30   33 opamp
r61     29 31 622.0
c61     29 31 0.001e-6
r62     31 32 781.0
c62     32 0 0.001e-6
r63     29 30 1000.0
r64     30 33 5514.0
rl6     33 0  50.0
*****************************
* ideal op amp w/ 100k gain and
* one-pole roll off at approx. 1 MHz
```

```
*****************************
.subckt opamp non inv out
rin non inv 1.0e6
egain 1 0   non inv 100.0e3
ropen 1 2   5.0e3
copen 2 0   30.0e-12
eout  3 0   2 0 1
rout  3 out 50
.ends opamp
*****************************
.ac dec 100 1.0 3e6
.end
```

C.2 90° Phase-Difference Network – AD846 Op-Amp Model

```
*****************************
*
* 12 pole I & Q all-pass network
* fa=100 Hz fb=1 MHz DKA 4/19/94
* Analog Devices AD846 op amp model;
* "real" chip R & C values
* (same basic circuit as IQ2C_12P.CIR
* with some diagnostic additions and
* output attenuator networks)
*
* I and Q outputs drive 50 ohms
*
* Q network has 0.95 attenuation on output
*
*****************************
vin 1 0 0.0 ac 1.4 0.0
vpos 100 0 dc 18.0
vneg 101 0 dc -18.0
*   vs1 and vs2 are pos. and neg.
*   supply current monitors
vs1   100 90
vs2   101 91
*****************************
* op amp 1 (I ckt)
*****************************
xamp1 3  2   90 91 5 50 AD846
r11    1  3   309.0
c11    1  3   0.001e-6
r12    3  4   169.0
c12    4  0   0.001e-6
r13    1  2   1000.0
r14    2  5   4120.0
*****************************
* op amp 2 (I ckt)
*****************************
xamp2 7  6   90 91 9 51 AD846
```

```
r21    5  7  768.0
c21    5  7  0.01e-6
r22    7  8  1370.0
c22    8  0  0.01e-6
r23    5  6  1000.0
r24    6  9  6650.0
*****************************
* op amp 3 (I ckt)
*****************************
xamp3 11 10 90 91 13 52 AD846
r31    9  11 3240.0
c31    9  11 0.1e-6
r32   11 12 4120.0
c32   12  0 0.1e-6
r33    9  10 1000.0
r34   10 13 5490.0
rs3   13 14 105.0
rp3   14 0  2.0e3
rl3   14 0  50.0
*****************************
* op amp 4 (Q ckt)
*****************************
xamp4 23 22 90 91 25 53 AD846
r41    1  23 7500.0
c41    1  23 0.2e-6
r42   23 24 4120.0
c42   24 0  0.2e-6
r43    1  22 1000.0
r44   22 25 4120.0
*****************************
* op amp 5 (Q ckt)
*****************************
xamp5 27 26 90 91 29 54 AD846
r51   25 27 1820.0
c51   25 27 0.01e-6
r52   27 28 3320.0
c52   28 0  0.01e-6
r53   25 26 1000.0
```

```
r54    26 29 6650.0
*****************************
* op amp 6 (Q ckt)
*****************************
xamp6 31 30 90 91 33 55 AD846
r61    29 31 619.0
c61    29 31 0.001e-6
r62    31 32 787.0
c62    32  0 0.001e-6
r63    29 30 1000.0
r64    30 33 5490.0
rs6    33 34 105.0
rp6    34 0  2.0e3
rl6    34 0  50.0
*****************************
* AD846 SPICE Macro-model 1/91, Rev. A
*                            JCB / PMI
*
* Copyright 1990 by Analog Devices, Inc.
*
* Refer to "README.DOC" file for
* License Statement.  Use of this model
* indicates your acceptance with the terms
* and provisions in the License Statement.
*
* Node assignments
*               non-inverting input
*               | inverting input
*               | | positive supply
*               | | | negative supply
*               | | | | output
*               | | | | | compensation pin
*               | | | | | |
.SUBCKT AD846  1 2 99 50 28 12
*
* INPUT STAGE
*
R1    99 8     1E3
```

49

```
R2    10 50      1E3
V1    99  9      1.7
D1     9  8      DX
V2    11 50      1.7
D2    10 11      DX
I1    99  5      275E-6
I2     4 50      275E-6
Q1    50  3  5   QP
Q2    99  3  4   QN
Q3     8  6 30   QN
Q4    10  7 30   QP
R3     5  6      300E3
R4     4  7      300E3
*
* INPUT ERROR SOURCES
*
BG1   99  1      I=3E-6 + (5E-9 * V(1,22))
BG2   99 30      I=150E-9 + (5E-9 * V(1,22))
VOS    3  1      25E-6
LS1   30  2      50E-9
CS1   99  2      2.0E-12
CS2   99  1      2.0E-12
*
EREF 97  0      22  0  1
*
* TRANSCONDUCTANCE STAGE
*
R5    12 97      200E6
C3    12 97      5.5E-12
G1    97 12      99  8  1E-3
G2    12 97      10 50  1E-3
V3    99 13      1.3
V4    14 50      1.3
D3    12 13      DX
D4    14 12      DX
*
* POLE AT 100 MHZ
*
```

```
R8   17 97      1E6
C4   17 97      1.59E-15
G4   97 17      12 22  1E-6
*
* POLE AT 100 MHZ
*
R9   18 97      1E6
C5   18 97      1.59E-15
G5   97 18      17 22  1E-6
*
* POLE AT 200 MHZ
*
R10 19 97       1E6
C6   19 97      0.795E-15
G6   97 19      18 22  1E-6
*
* POLE AT 500 MHZ
*
R11 20 97       1E6
C7   20 97      0.318E-15
G7   97 20      19 22  1E-6
*
* POLE AT 500 MHZ
*
R12 21 97       1E6
C8   21 97      0.318E-15
G8   97 21      20 22  1E-6
*
* OUTPUT STAGE
*
ISY 99 50       3.55E-3
R13 22 99       16.67E3
R14 22 50       16.67E3
R15 27 99       60
R16 27 50       60
L2   27 28      1E-8
G9   25 50      21 27  16.67E-3
G10 26 50       27 21  16.67E-3
```

```
G11 27 99      99 21  16.67E-3
G12 50 27      21 50  16.67E-3
V5  23 27      1.35
V6  27 24      1.35
D5  21 23      DX
D6  24 21      DX
D7  99 25      DX
D8  99 26      DX
D9  50 25      DY
D10 50 26      DY
*
* MODELS USED
*
.MODEL QN   NPN(BF=1E9 IS=1E-15 VAF=16.3E3)
.MODEL QP   PNP(BF=1E9 IS=1E-15 VAF=16.3E3)
.MODEL DX   D(IS=1E-15)
.MODEL DY   D(IS=1E-15 BV=50)
.ENDS AD846
****************************
.ac dec 100 1.0 3e6
.end
```

Distribution

Admnstr
Defns Techl Info Ctr
Attn DTIC-OCP
8725 John J Kingman Rd Ste 0944
FT Belvoir VA 22060-6218

Hdqtrs Dept of the Army
Attn DAMO-FDQ MAJ M McGonagle
400 Army Pentagon
Washington DC 20310-0460

US ARDEC
Attn AMSTA-AR-AEF-A S Weiss
Adelphi MD 20783-1197

US Army Natl Ground Intllgnc Ctr
Attn T Caldwell
220 Seventh St NE
Charlottesville VA 22901-5396

Nav Rsrch Lab
Attn Code 6793 S Gold
Attn Code 6840 R Parker
Attn Code 6841 B Levush
Attn Code 6843 B Danly
4555 Overlook Ave SW
Washington DC 20375

NSWC Indian Head Div
Attn Code R13 D Tasker
10901 New Hampshire Ave
Silver Spring MD 20903-5640

NSWC Indian Head Div
Detonation Physics Div
Attn Code 9220 B Wilson
Silver Spring MD 20904-5640

US Air Force Phillips Lab
Attn PL/WSR F J Agee
3550 Aberdeen Ave SE
Kirtland AFB NM 87117-5776

Lawrence Livermore Natl Lab
Attn L-153 D Goerz
Attn L-35 J B Chase
PO Box 808
Livermore CA 94550

Los Alamos Natl Lab
Attn MS H851 M Fazio
Los Alamos NM 87545

Univ of CA
Dept of Applied Sci & Elect & Computer
   Engrg
Attn N C Luhmann JR
228 Walker Hall
Davis CA 95616

University of Maryland
Inst for Plasma Rsrch
Attn V Granatstein
Attn Y Carmel
Attn J Rodgers
College Park MD 20742-3511

University of New Mexico
Elect & Computer Engrg Dept
Attn E Schamiloglu
Albuquerque NM 87131

Berkeley Rsrch Associates
Attn J Golden
PO Box 852
Springfield VA 22150

CDR NGIC
Attn IANG-TMA R Rains
220 Seventh Stret NE
Charlottesville VA 22902-5396

CPI
Beverly Microwave Div
Attn T A Treado
150 Sohier Rd
Beverly MA 01915-5595

Distribution

Hughes Rsrch Lab
Attn MS RL58 J Butler
3011 Malibu Canyon Rd
Malibu CA 90265

Lockheed Martin Corp
Advanced Dev Operations
Attn D Furuno
15250 Ave of Science
San Diego CA 92128

Mission Rsrch Corp
Attn M Bollen
Attn R Smith
8560 Cinderbed Rd Ste700
Newington VA 22122

Northrop Grumman ES1D-RM
Attn C Armstrong
600 Hicks Rd Rm H6402
Rolling Meadows IL 60008

Titan-AIT
Attn R B Miller
PO Box 9254
Albuquerque NM 87119-9254

US Army Rsrch Lab
Attn AMSRL-SL-EP G Palomino
White Sands Missile Range NM 88002-5513

US Army Rsrch Lab
Attn AMSRL-CS-AL-LL Tech Lib (3 copies)
Attn AMSRL-CS-AL-TA Mail & Records
  Mgmt
Attn AMSRL-CS-AL-TP Tech Pub (5 copies)
Attn AMSRL-IS P Emmerman
Attn AMSRL-IS-PA S Ho
Attn AMSRL-IS-TA A Mark
Attn AMSRL-SE-D E Scannell
Attn AMSRL-SE-DP C Lazard
Attn AMSRL-SE-DP D Judy
Attn AMSRL-SE-DP G Huttlin
Attn AMSRL-SE-DP M Litz
Attn AMSRL-SE-DP A Bromborsky
Attn AMSRL-SE-DP D Abe (20 copies)
Attn AMSRL-SE-DP H Brandt
Attn AMSRL-SE-DP R A Kehs
Attn AMSRL-SE-DP R del Rosario
Attn AMSRL-SE-DP A Bromborsky
Attn AMSRL-WT-ND H Brisker
Attn AMSRL-WT-ND L Libelo
Attn AMSRL-WT-NF B Benwell
Attn AMSRL-WT-NF D DeTroye
Attn AMSRL-WT-NF J Tatum
Attn AMSRL-WT-NF M Berry
Attn AMSRL-WT-NF R Sebol
Attn AMSRL-WT-NF T Turner
Adelphi MD 20783-1197

REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE September 1996	3. REPORT TYPE AND DATES COVERED Final, from Feb 1995 to Feb 1996

4. TITLE AND SUBTITLE Design and Testing of an Interference Modulator for DEW Applications	5. FUNDING NUMBERS PE: 622120
6. AUTHOR(S) David K. Abe, Andrea K. Mark, Alan Bromborsky, Carl J. Lazard, and Romeo del Rosario	

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory AMSRL-WT-NH 2800 Powder Mill Road Adelphi, MD 20783-1197	8. PERFORMING ORGANIZATION REPORT NUMBER ARL-TR-1049
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory 2800 Powder Mill Road Adelphi, MD 20783-1197	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

AMS code: 622120.1400011
ARL PR: 6WN162

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.	12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The interference modulator (IM) is a microwave source concept that can produce a broadband, amplitude modulated radio frequency (rf) waveform with independently tunable carrier and modulation frequencies. In its simplest form, the IM is a two-channel device that can produce an amplitude modulated signal at 50-percent duty by either generating and beating two electromagnetic waves together in the far field or by combining the two signals with a four-port hybrid network. With additional channels, an IM can be made to synthesize more complex waveforms and duty factors. A general methodology for the design of an n-channel ($n \geq 2$) IM is developed. As a validation of the design methodology, a two-channel prototype IM was constructed. The design, fabrication, and testing of the two-channel prototype are described in detail. Experiments carried out with the prototype IM have successfully demonstrated the IM concept, producing a source with a modulation envelope that is continuously tunable over four decades (from 100 Hz to 1 MHz), with a carrier frequency that is continuously tunable from 800 MHz to 2 GHz.

14. SUBJECT TERMS interference modulator, single sideband	15. NUMBER OF PAGES 59
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102